

Overview:

The project aims to implement the requirements mentioned in the document and to address the necessary refactorization.

As technologies for the implementation, it has been used as following:

- .Net Core 3.1 for API development
- MSSQL database for storage system
- EF Core as an ORM to connect to the storage systems
- Swagger for API interaction and possible documentation
- Fluent Validation for input validation
- Docker for containerization support

The projects' structure follows clean architecture with elements from design driven development.

To achieve this CQRS and Mediator patters have been used.

High level architecture:

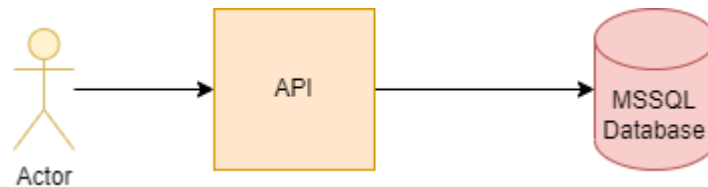


Fig 1.0 High Level Diagram

API Structure:

The API is structured as following:

Presentation -> Web API

Core -> Application + Domain

Infrastructure -> Persistence + Database

Tests -> Unit Tests

Data Flow:

- The data flow starts in the web API project.
- When a specific endpoint is called a specific query or command is instantiated and called
- The command and query data are validated using query validators defined on the same folder level as the queries or commands.
- In the Application project this can be identified in the **UseCases** folder. (See Fig. 2.0)
- The database schemas and the EF Core configurations can be found in the Persistence project and Database Project. (See Fig 3.0)

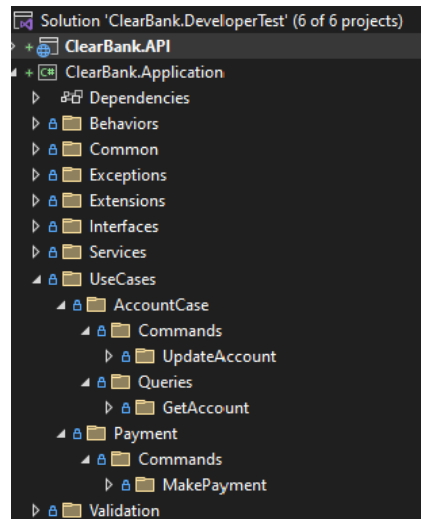


Fig 2.0 Application Project and Use Cases

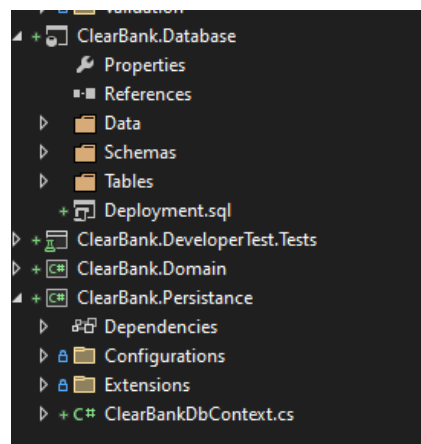


Fig 3.0 Persistence and Database Projects

- For each project there is a dependency injection extension which is called in the end in the startup project to glue all together.

Setup Database

The prerequisite will be to have installed an MSSQL server installed on the system.

After this step is completed to setup the database you can do this by pressing **Publish** button from the Database project in Visual Studio.

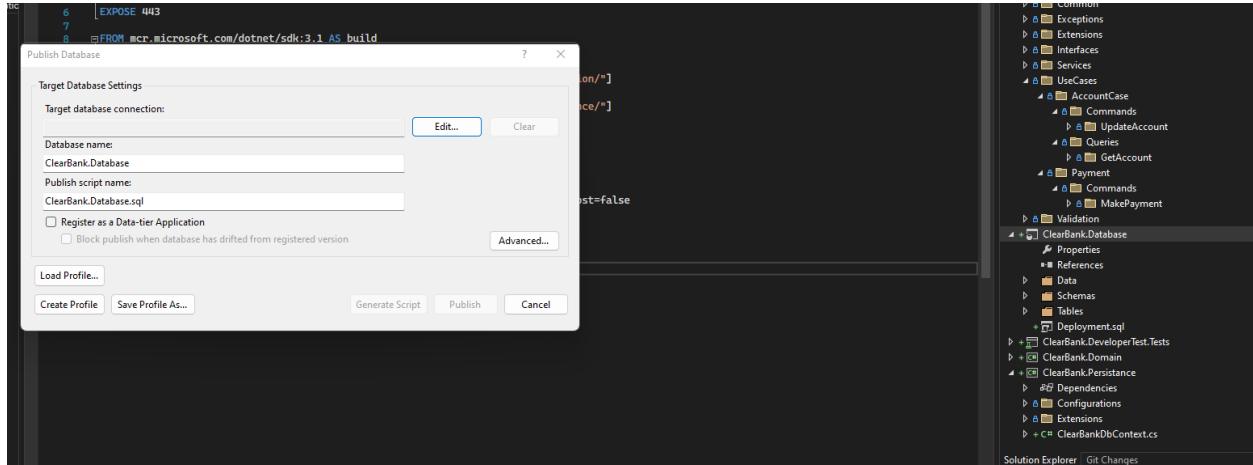


Fig 4.0 Publish Database

After that you will have to select the SQL Server to which you want to run the publish against.

Start Project:

When the database is set you can start and run the project and the Swagger UI will appear.

All the requests can be made from UI.

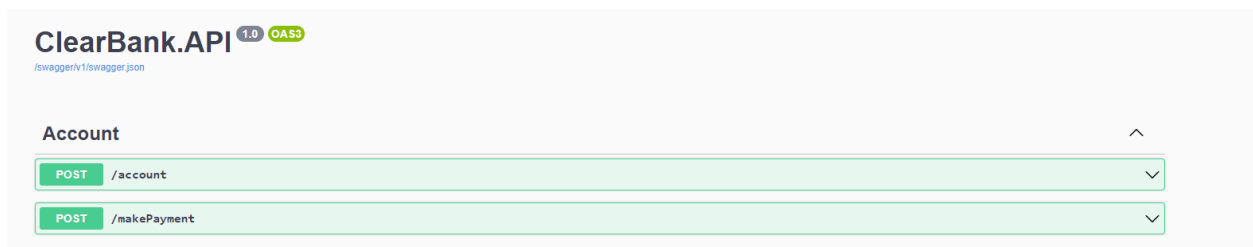


Fig 5.0 Swagger UI